Van Leeuwen & Van Leeuwen
Attorneys at Law
6123 Pebble Garden Court
Austin, Texas 78739
Phone # 512-301-6738
FAX # 512-301-6742

DATE: January 10, 2006

Number of Pages to Follow (including this cover sheet) 32

SEND TO:	United States Patent Office
Examiner:	**Mitchell, Jason D.**
Group Art Unit:	**2124**
Tel No:	571-272-3728
Fax #:	703-872-9306

FROM:	Joseph T. Van Leeuwen **Van Leeuwen & Van Leeuwen** **Registered Patent Attorneys** 6123 Pebble Garden Court Austin, Texas 78739
Tel No:	512-301-6738
Fax No.	512-301-6742

Docket No. RSW920010187US1 **Serial No. 10/047,312** Atty: GRW / JVL

Applicant: Barker, et al.

X Transmittal Letter (2 copies)	X Certificate of Facsimile (incl. w/Amendment)
____ Amendment (__ pages)	____ Notice of Appeal
____ Amendment AF	X Appeal Brief (29 pages)
____ Ext. of Time	____ Reply Brief
____ IDS Statement	____ Change of Address
____ Other	

Deposit Acct. No. 09-0461

Fees: Amendment _____ Notice of Appeal _____ Appeal Brief $500 Other _____
File: **IBM-R112**

COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, VA 22313-1450

IBM DOCKET NO. RSW920010187US1
DATE: January 10, 2006

Application Serial No.: 10/047,312

Sir:

Assignee Name: International Business Machines Corporation
Assignee Residence: Armonk, New York

Transmitted herewith for filing is the Patent Application of:

Inventors: Barker, et al.
For: System and Method for Packaging and Installing Management Models with Specific Console Interfaces

Enclosed are:

_X_ Appeal Brief ($500).

Any additional filing fees have been calculated as shown below:

_X_ Please charge my Deposit Account No. 09-0461 in the amount of $500.00. A duplicate copy of this sheet is enclosed.

_X_ The Commissioner is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to Deposit Account No. 09-0461. A duplicate copy of this sheet is enclosed.

_X_ Any additional filing fees required under 37 CFR Sect. 1.16.

_X_ Any patent application processing fees under 37 CFR Sect. 1.17.

_X_ No extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and the undersigned hereby authorizes the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0461.

Respectfully submitted,

By _____
Joseph T. Van Leeuwen, Reg. No. 44,383
Van Leeuwen & Van Leeuwen
Attorneys for Applicant
Telephone: (512) 301-6738
Facsimile: (512) 301-6742

Atty Ref. No. IBM-R112

RECEIVED
CENTRAL FAX CENTER
JAN 10 2006

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Barker, et. al.	§ Group Art Unit: 2193
	§ Examiner: Mitchell, Jason D.
Serial No.: 10/047,312	§ Attorney Docket No.
Filed: January 14, 2002	§ RSW920010187US1
Title: System and Method for Packaging and Installing Management Models with Specific Console Interfaces	§ IBM Corporation § Software Group IP Law T81/503 § 3039 Cornwallis Road § Research Triangle Park, NC 27709

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Certificate of Mailing or Transmission
I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 or facsimile transmitted to the U.S. Patent and Trademark Office on the date shown below.

Joseph T. Van Leeuwen — 1/10/2006 Date

APPELLANTS' BRIEF

Sir:

A. INTRODUCTORY COMMENTS

This brief is filed in support of the previously filed Notice of Appeal, filed in this case on November 14, 2005, which appealed from the decision of the Examiner dated August 11, 2005 finally rejecting claims 1-25. Please charge the required fee for this Appeal Brief to IBM Corporation Deposit Account No. 09-0461.

The two-month deadline for filing this Appeal Brief is January 16, 2006 (as January 14 falls on a Saturday), therefore, no extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and the undersigned hereby authorizes the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0461.

PAGE 4/32 * RCVD AT 1/10/2006 8:16:54 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-6/25 * DNIS:2738300 * CSID:512 301 6742 * DURATION (mm-ss):11-54
01/11/2006 KBETEMA1 00000101 090461 10047312
01 FC:1402 500.00 DA

B. REAL PARTY IN INTEREST

The real party in interest in this appeal is International Business Machines Corporation, which is the assignee of the entire right, title, and interest in the above-identified patent application.

C. RELATED APPEALS AND INTERFERENCES

With respect to other prior or pending appeals, interferences, or judicial proceedings that are related to, will directly affect, be directly affected by, or have a bearing on the Board's decision in the pending appeal, the following appeals have been filed in applications noted as related and known to Appellants, Appellants' legal representative, or assignee:

1. 10/047,792, Appellants' Appeal Brief filed November 21, 2005; and

2. 10/046,940, Appellants' Appeal Brief filed July 28, 2005, Examiner's Answer mailed October 19, 2005, Appellants' Reply Brief filed December 19, 2005.

D. STATUS OF CLAIMS

1. Total number of claims in application

There are 25 claims pending. Seven claims are independent claims (1, 8, 15, and 22-25), and the remaining claims are dependent claims.

2. Status of all claims in application

- Claims canceled: none.
- Claims withdrawn from consideration but not canceled: none
- Claims pending: 1-25
- Claims allowed: None
- Claims rejected: 1-25

3. Claims on appeal

The claims on appeal are: claims 1-25.

E. STATUS OF AMENDMENTS

All amendments have been entered in this case. No amendments have been made to the claims after the Final Office Action.

F. SUMMARY OF CLAIMED SUBJECT MATTER

Appellants provide a concise summary of the claimed subject matter as follows. Claims 1, 8, 15 and 22-25 are independent claims. Note that claims 1, 22, and 23 are method claims, claims 8 and 24 are information handling system claims, and claims 15 and 25 are computer program product claims. Independent claims 15 and 25 include means plus function limitations that correspond to the method steps set forth in independent claim 1 and 22, respectively. An information handling system capable of implementing Appellants' invention, as claimed in independent claims 8 and 24, is shown in Figure 18, element , and described in Appellants' specification on page 45, line 10 through page 46, line 24. Support for independent computer program product claims 15 and 25 is described in Appellants' specification on page 46, line 25 through page 47, line 14. In addition, support for each of the method steps and means plus function limitations of the independent claims are discussed below. The specific citations to Appellants' Figures and Specification are meant to be exemplary in nature, and do not limit the scope of the claims In particular, the citations below do not limit the scope of equivalents as provided under 35 U.S.C. § 112, sixth paragraph.

In one aspect of Appellants' invention, claim 1 claims a method of packaging management data adapted to interoperate with one or more management consoles (see, e.g., Figure 1, elements 125 and 130 , specification page 10, line 8 through page 13, line 6) receiving one or more console identifiers, each of the console identifiers corresponding to one of the

management consoles (see, e.g., Figure 6, element 650, specification page 21, line 18 through page 23, line 7);

retrieving one or more plug-in code files, each of the plug-in code files derived from the management data and each adapted to interface with one of the management consoles (see, e.g., Figure 2, elements 245 to 270 , specification page 13, line 7 through page 15, line 5);

retrieving one or more display panel files derived from the management data (see, e.g., Figure 2, elements 230 and 270 , specification page 13, line 7 through page 15, line 5); and

writing the plug-in code files and the display panels to a distribution medium (see, e.g., Figure 2, elements 270 and 275 , specification page 13, line 7 through page 15, line 5).

In another aspect, the claimed invention is a method, information handling system, and computer program product for packaging management data adapted to interoperate with one or more management consoles, wherein the management data includes a common information model managed object format file (see, e.g., Figure 5, element 500, specification page 20, line 9 through page 21, line 17), the method comprising:

receiving one or more console identifiers, each of the console identifiers corresponding to one of the management consoles (see, e.g., Figure 6, element 650, specification page 21, line 18 through page 23, line 7);

retrieving one or more plug-in code files, each of the plug-in code files derived from the management data and each adapted to interface with one of the management consoles (see, e.g., Figure 2, elements 245 to 270 , specification page 13, line 7 through page 15, line 5);

retrieving one or more display panel files derived from the management data (see, e.g., Figure 2, elements 230 and 270 , specification page 13, line 7 through page 15, line 5);

retrieving one or more translation files derived from the management data, each of the translation files corresponding to at least one national language (see, e.g., Figure 2, elements 260 to270, specification page 13, line 7 through page 15, line 5); and

writing the translation files, the plug-in code files and the display panels to a distribution medium (see, e.g., Figure 2, elements 270 and 275 , specification page 13, line 7 through page 15, line 5).

In another aspect, the claimed invention is a method of packaging management data adapted to interoperate with one or more management consoles, wherein the management data includes a common information model managed object format file (see, e.g., Figure 5, element 500, specification page 20, line 9 through page 21, line 17), the method including receiving one or more console identifiers, each of the console identifiers corresponding to one of the management consoles (see, e.g., Figure 6, element 650, specification page 21, line 18 through page 23, line 7);

retrieving one or more plug-in code files, each of the plug-in code files derived from the management data and each adapted to interface with one of the management consoles (see, e.g., Figure 2, elements 245 to 270 , specification page 13, line 7 through page 15, line 5);

retrieving one or more display panel files derived from the management data (see, e.g., Figure 2, elements 230 and 270 , specification page 13, line 7 through page 15, line 5);

retrieving one or more translation files derived from the management data, each of the translation files corresponding to at least one national language (see, e.g., Figure 2, elements 260 to270, specification page 13, line 7 through page 15, line 5);

retrieving one or more plug-in runtime algorithms, each of the algorithms corresponding to one of the console identifiers;

generating a console plug-in code file for each of the console identifiers (see, e.g., Figure 2, element 245, specification page 13, line 7 through page 15, line 5);

compiling each of the generated console plug-in files, the compiling resulting in an executable entity adapted to interface with the management console corresponding to the console identifier (see, e.g., Figure 2, element 270, specification page 13, line 7 through page 15, line 5);

writing the translation files, the compiled plug-in code files and the display panels to a distribution medium (see, e.g., Figure 2, elements 270 and 275 , specification page 13, line 7 through page 15, line 5).

Support for each of Appellants' means plus function limitations set forth in dependent claims is provided below. Note that general support for an information handling system and

computer program product is discussed above. The specific citations to Appellant's Figures and Specification are meant to be exemplary in nature, and do not limit the scope of the claims, as provided under 35 U.S.C. § 112, sixth paragraph.

Claim 17: means for retrieving one or more translation files derived from the management data, each of the translation files corresponding to at least one national language (see, e.g., Figure 2, elements 260-268 , specification page 13, line 7 through page 15, line 5); and

means for writing the translation files to the distribution medium (see, e.g., Figure 2, element 270, specification page 13, line 7 through page 15, line 5).

Claim 19: means for retrieving one or more plug-in runtime algorithms, each of the algorithms corresponding to one of the console identifiers (see, e.g., Figure 2, elements 245 and 250 , specification page 13, line 7 through page 15, line 5);

means for generating a console plug-in code file for each of the console identifiers (see, e.g., Figure 2, element , 245 specification page 13, line 7 through page 15, line 5); and

means for compiling each of the generated console plug-in files, the compiling resulting in an executable entity adapted to interface with the management console corresponding to the console identifier (see, e.g., Figure 2, element 270, specification page 13, line 7 through page 15, line 5).

Claim 20: means for loading the distribution medium onto a computer system (see, e.g., Figure 6, elements 605 and 610, specification page 21, line 18 through page 23, line 7);

means for displaying a name corresponding to each of the management consoles in a selection display (see, e.g., Figure 6, element 630 and 640, specification page 21, line 18 through page 23, line 7);

means for receiving one or more selections from a user, each of the selections corresponding to one of the management consoles (see, e.g., Figure 6, element 650 , specification page 21, line 18 through page 23, line 7);

means for copying the plug-in code files corresponding to the selected management consoles from the distribution medium to a nonvolatile storage device accessible by the

computer system (see, e.g., Figure 6, element 660, specification page 21, line 18 through page 23, line 7);

means for copying the display panel files from the distribution medium to a nonvolatile storage device accessible by the computer system (see, e.g., Figure 6, element 660, specification page 21, line 18 through page 23, line 7); and

means for registering each of the plug-in code files with one or more installed management consoles, wherein the installed management consoles are installed on the computer system (see, e.g., Figure 6, element 670, specification page 21, line 18 through page 23, line 7).

Claim 21: means for invoking one of the installed management consoles (see, e.g., Figure 7, element 705, specification page 23, line 8 through page 25, line 22);

means for receiving a console selection from a user (see, e.g., Figure 7, element 715, specification page 23, line 8 through page 25, line 22); and

means for displaying a display panel corresponding to one of the display panel files in response to the received selection (see, e.g., Figure 7, element 770, specification page 23, line 8 through page 25, line 22).

G. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1, 3-8, 10-15, 17-21 and 25 stand rejected under 35 U.S.C. § 102(e) as being anticipated and therefore unpatentable over U.S. Patent No. 6,311,321 to Agnihotri et al. (hereinafter "Agnihotri"). Claims 2, 9, 16, and 22-24 stand rejected under 35 U.S.C. § 103 as being obvious, and therefore unpatentable, over Agnihotri in view of a reference entitled "Common Information Model (CIM) Specification v. 2.2," (hereinafter "CIM" or "the CIM reference").

H. ARGUMENTS

I. 35 U.S.C. § 102, Alleged Anticipation

1. Independent Claims 1, 8, 15 and 25

The Examiner rejected claims 1, 3-8, 10-15, 17-21, and 25 under 35 U.S.C. § 102 as being anticipated by U.S. Patent No. 6,311,321 (hereinafter, "Agnihotri"). For the reasons set forth below, Agnihotri fails to anticipate any of these claims.

With respect to claim 1, the rejection of which is representative of that of independent claims 8 and 15 claims, the Examiner stated:

> Agnihotri discloses receiving one or more console identifiers, each of the console identifiers corresponding to one of the management consoles (col. 5, lines 3-7 'provides the user the option to select one console'); retrieving one or more plug-in code files, each of the plug-in code files derived from the management data (col. 4, lines 50-55 'applet components') and each adapted to interface with one of the management consoles (col. 4, lines 50-55 'specific to the Enterprise management console'); retrieving on or more display panel files derived from the management data (col. 4, lines 58-63 'Install interface'); and writing the plug-in code files and the display panels to a distribution medium (col. 4, lines 32-35 'a software module provided on a tangible medium').

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond,* 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). Agnihotri fails to anticipate the presently claimed invention because it fails to show all of the elements of the claimed invention.

The rejected independent claims, 1, 8, 15, and 25, all recite "plug-in code files *derived from the management data*" and "display panel files *derived from the management data*" (emphasis added). These features are not taught or suggested by Agnihotri. While the Examiner

has cited various excerpts from Agnihotri that the Examiner argues teach these claim elements, Agnihotri fails to teach or otherwise describe or suggest the origin of the "applet components" and "install interface" the Examiner refers to.

For instance, the Examiner cites to col. 4, lines 50-55 of Agnihotri as teaching the claimed "plug-in code files" by describing "applet components." This excerpt is reproduced below:

> The Install module **210** may be used to provide a file based interface to applets for installing applet components, such as the applet.exe and dynamics link libraries (DLLs), and components specific to the Enterprise management console such as class object definitions (in the case of CA Unicenter console). [col. 4, lines 50-55].

While Appellants fundamentally disagree with the Examiner's equating "applet components" to the claimed "plug-in code files," even if one assumes, for the sake of argument, that the claimed "plug-in code files" and Agnihotri's "applet components" could be equated in this fashion, Agnihotri would still fail to teach or suggest the present invention as claimed, since Agnihotri fails to teach or suggest that the plug-in code files are *derived from management data*, as recited in independent claims 1, 8, 15, and 25. In fact, Agnihotri fails to provide any indication as to the origin of its "applet components" at all.

Likewise, col. 4, lines 58-63 of Agnihotri fails to teach or suggest display panel files *derived from management data:*

> As shown in **FIG. 3**, the Install module **210** comprises an Install framework **212**, and install interface DLL **214**, and a plurality of comprehensive generic interface for installation of applets into existing Enterprise management consoles such as HP OpenView, CA Unicenter and other MConsoles via corresponding HP OpenView Libraries **302**, and CA Unicenter Libraries **312** and other MConsole #N Interface Libraries **322**. [col. 4, lines 58-67].

The present invention, on the other hand, is specifically directed to the derivation of plug-in code and display panels from management data. In a preferred embodiment, these plug-in

code files and display panel files are specifically derived from management definition objects, such as from an MOF (Management Object Format) file, as in the excerpt from Appellants' specification reproduced below:

> Management definition object **220**, such as a MOF, is read and evaluated (step **215**) to determine the panels, plug-in code, and NLS data needed to process. [Appellants' specification, p. 13, lines 11-14].

As the Summary section of the present application explicitly points out, the present invention is specifically directed to a plug-in *builder* process. Hence, the origins of the plug-in code files and display panels, as recited in the claims, are important features of the invention and cannot properly be ignored. Indeed, the very first sentence of Appellants' Summary reads:

> It has been discovered that a plug-in builder process can be performed using a management definition object, such as a CIM MOF file, to allow objects included in the management definition object to be accessible through a variety of management consoles. [Appellants' specification, p. 5, lines 3-6].

Hence, while the presently rejected claims recite "plug-in code files derived from the management data" and "display panel files derived from the management data," the cited Agnihotri reference fails to teach or suggest these features. Therefore Agnihotri fails to teach all elements of the claimed invention and, consequently, fails to anticipate the invention as recited in claims 1, 8, 15, and 25

In addition, with respect to independent claim 25, Agnihotri fails to teach or suggest the claimed feature of "retrieving one or more translation files derived from the management data, each of the translation files *corresponding to at least one national language*" (emphasis added). Although the Examiner cites an excerpt from Agnihotri that makes a passing reference to "language," the excerpt makes no mention of retrieving any translation files corresponding to any national language. In fact, from the context, it appears that by "language," Agnihotri is referring to a computer programming language, such as C or Pascal, rather than a "national" or "natural" language used for human communication.

> The Install Interface DLL **214** may store generic instruction sets (i.e., interface programs) for supporting the installation process, including providing console information such as installation directory, language, console version, etc. Individual instructions sets (in bold print) and textual comments may be written in any of the C-family (e.g., C or C++) code language. However, other program languages included in the non-exhaustive list of Basic and Pascal may also be used. [col. 5, lines 43-51].

Thus, for this additional reason, Agnihotri fails to anticipate independent claim 25.

For the foregoing reasons, Appellants respectfully submit that Agnihotri does not teach or suggest all of the limitations of the present invention as recited in independent claims 1, 8, 15, and 25. Appellants maintain that these claims are patentable over Agnihotri. Consequently, Appellants respectfully request that the Board ***REVERSE*** the rejections of claims 1, 8, 15, and 25.

2. Dependent Claims Rejected Under § 102

Claims 3-7, 10-14, and 17-21 are dependent claims that depend from independent claims 1, 8, and 15. Appellants have already demonstrated claims 1, 8, and 15 to be in condition for allowance. Appellants respectfully submit that claims 3-7, 10-14, and 17-21 are also allowable, at least by virtue of their dependency on allowable claims. Moreover, these claims are further allowable on separate grounds as discussed below.

2a. Dependent Claims 3, 10, and 17

Dependent claims 3, 10 and 17 depend on claims 1, 8, and 15, respectively and add further limitations of "retrieving one or more translation files derived from the management data, each of the translation files corresponding to at least one national language," and "writing the translation files to the distribution medium." In the Final Office Action, the Examiner contends that these further limitations are taught by Agnihotri citing col. 4, lines 50-55.

> The Install module 210 may be used to provide a file based interface to applets for installing applet components, such as the applet.exe and dynamics link libraries

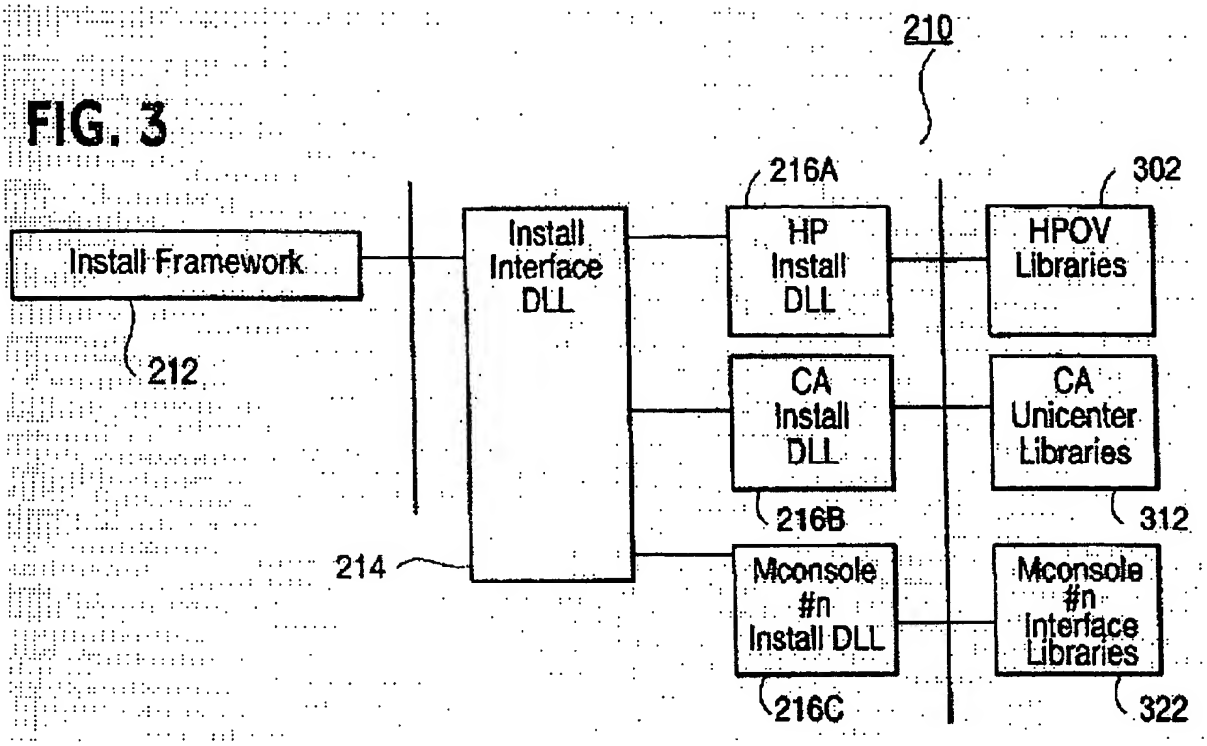
> (DLLs), and components specific to the Enterprise management console such as class object definitions (in case of CA Unicenter console).

Nowhere in the cited section, or elsewhere in Agnihotri, does Agnihotri teach or suggest retrieving "national language" translation files. As previously discussed, Agnihotri's discussion of languages appears directed to computer languages such as C or Pascal, and not directed to "national" or "natural" languages. The cited section does not teach or describe anything to do with national languages or retrieving translation files, as claimed by Appellants. Instead, the cited section simply describes how Agnihotri's install module may be used for installing applet (computer code) components, with no regard whatsoever for national languages. Because Agnihotri does not teach or suggest the limitations found in claims 3, 10, and 17, these claims are allowable over Agnihotri and Appellants respectfully request that the Board ***REVERSE*** these rejections.

2b. Dependent Claims 4, 11, and 18

Dependent claims 4, 11 and 18 depend on claims 1, 8, and 15, respectively and add a further limitation "wherein each of the display panel files is adapted to operate with a plurality of the management consoles." In the Final Office Action, the Examiner contends that this further limitation is taught by Agnihotri citing col. 4, lines 58-63 as teaching a "generic interface." However, this section of Agnihotri does not teach or suggest a display panel that is adapted to operate with a plurality of management consoles. Instead, Agnihotri teaches separate display interfaces for use with various management consoles. The cited section and Agnihotri's Figure 3 are reproduced below:

> As shown in FIG. 3, the Install module 210 comprises an Install framework 212, an Install interface DLL 214, and a plurality of console install DLLs 216A, 216B, 216C for providing a comprehensive generic interface for installation of applets into existing Enterprise management consoles such as HP OpenView, CA Unicenter and other MConsoles via corresponding HP OpenView Libraries 302, and CA Unicenter Libraries 312 and other MConsole #N Interface Libraries 322.



Agnihotri specifically teaches that the "plurality of console install DLLs 216A, 216B, 216C" are used to provide the interfaces for installation. As shown in Agnihotri's Figure 3, these DLLs are each directed towards a different management console. Consequently, Agnihotri teaches that the display panels are coded in the separate DLLs and are not "adapted to operate with a plurality of the management consoles," as taught and claimed by Appellants in claims 4, 11, and 18. Because Agnihotri does not teach or suggest the limitations found in claims 4, 11, and 18, these claims are allowable over Agnihotri and Appellants respectfully request that the Board ***REVERSE*** these rejections.

2c. Dependent Claims 5, 12, and 19

Dependent claims 5, 12, and 19 depend on claims 1, 8, and 15, respectively and add further limitations of "retrieving one or more plug-in runtime algorithms, each of the algorithms corresponding to one of the console identifiers," "generating a console plug-in code file for each of the console identifiers," and "compiling each of the generated console plug-in files, the

compiling resulting in an executable entity adapted to interface with the management console corresponding to the console identifier." In the Final Office Action, the Examiner contends that these further limitations are taught by Agnihotri citing 49 words of Agnihotri (col. 7, lines 3-8) as teaching ***every one*** of these limitations. As discussed below, this section of Agnihotri does not teach or suggest the limitations of claims 5, 12, and 19. This section of Agnihotri is reproduced below:

> The individual interface COM objects such as the HP Interface COM object 224, the CA Interface COM object 226 and MConsole #N interface COM object 228 may act as translators for respective Enterprise management consoles which may translate instructions from the MConsole Interface DLLs 222 into console specific commands.

In the cited section, Agnihotri is discussing interface COM objects that are used to ***translate*** instructions from one console (the MConsole Interface) to other consoles. Nowhere does Agnihotri teach or suggest "retrieving ... plug-in runtime algorithms" that correspond to one of the consoles, as taught and claimed by Appellants. Moreover, Agnihotri does not teach "compiling ... the generated console plug-in files ... resulting in an executable ... adapted to interface with the management console..." Instead, as discussed above, Agnihotri is simply teaching use of a module that translates console instructions from one console to another console. Because Agnihotri does not teach or suggest the limitations found in claims 5, 12, and 19, these claims are allowable over Agnihotri and Appellants respectfully request that the Board ***REVERSE*** these rejections.

2d. Dependent Claims 6, 13, and 20

Dependent claims 6, 13, and 20 depend on claims 1, 8, and 15, respectively and further limitations of "loading the distribution medium into a computer system," "displaying a name corresponding to each of the management consoles in a selection display," "receiving one or more selections from a user, each of the selections corresponding to one of the management consoles," "copying the plug-in code files corresponding to the selected management consoles from the distribution medium to a nonvolatile storage device accessible by the computer system," "*copying the display panel files* from the distribution medium to a nonvolatile storage

device accessible by the computer system," and "registering each of the plug-in code files with one or more installed management consoles, wherein the installed management consoles are installed on the computer system."

One of many differences between Appellants' claimed invention and the teachings of Agnihotri that Appellants' claimed invention differentiate between code use by consoles and display files used to display information to the user. Differentiating between program code and display files allows Appellants' claimed invention to provide a plurality of national language display panels that displays information in the user's language of choice. As previously discussed, Agnihotri does not teach or suggest providing national language display panels nor does Agnihotri teach or suggest providing display panels separately from the program code (applets), as taught and claimed by Appellants. The Final Office Action asserts that Agnihotri teaches Appellants' limitation of "copying the display panel files from the distribution medium to a nonvolatile storage device accessible by the computer system," citing col. 5, lines 15-18. However, this section of Agnihotri clearly only discusses installing the "applet" (code) and never teaches or suggests installing a separate display panel:

> After the inputs are confirmed by the user, the wizard application may proceed to install the component (***applet***) and make appropriate updates to the console to integrate the ***component (applet)***. (emphasis added)

Because Agnihotri does not teach or suggest installing "display files" separately from code files, as claimed in claims 5, 12, and 19, these claims are allowable over Agnihotri and Appellants respectfully request that the Board ***REVERSE*** these rejections.

2e. Dependent Claims 7, 14, and 21

Dependent claims 7, 14, and 21 depend on claims 6, 13, and 20, respectively (the allowability of which was discussed in the preceding section), and further limitations of: "invoking one of the installed management consoles," "receiving a console selection from a user," and "displaying a display panel corresponding to one of the display panel files in response to the received selection." As discussed in the preceding section, Agnihotri does not teach or

suggest separating display panels from code files. The Final Office Action contends that Appellants last limitation (displaying a display panel corresponding to one of the display panel files in response to the received selection) is taught by Agnihotri at col. 5, lines 10-12. However, this section of Agnihotri merely states that an "install file may contain a configuration file and an image file for graphical representation on the console." Importantly, neither this section, nor anywhere else in Agnihotri, teach or suggest displaying a display panel that is separate and apart from the program code as taught and claimed by Appellants. Accordingly, because Agnihotri does not teach or suggest at least this element of claims 7, 14, and 21 (nor the claims on which these claims depend), Appellants respectfully request that the Board ***REVERSE*** the Examiner's rejection of these claims.

As discussed above, the Agnihotri reference simply does not anticipate any of Appellants' claims rejected under 35 U.S.C. § 102. Accordingly, Appellants respectfully request that the Board ***REVERSE*** the Examiner's rejection of claims 1, 3-8, 10-15, 17-21, and 25 and that these claims be allowed to issue.

II. 35 U.S.C. § 103, Alleged Obviousness

The Examiner rejected claims 2, 9, 16, and 22-24 under 35 U.S.C. § 103 as being obvious in view of US 6,311,321 (Agnihotri et al.) 2001-10-30 and "Common Information Model (CIM) Specification v. 2.2" (CIM). As an initial matter, each of these claims either depends on claims rejected under § 102 or includes allowable limitations included in claims that were rejected under § 102 and, therefore, each of these claims is allowable for at least this reason.

The Office bears the burden of establishing a *prima facie* case of obviousness based on the prior art when rejecting claims under 35 U.S.C. § 103. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992). For an invention to be prima facie obvious, the prior art must teach or suggest all claim limitations. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). "All words in a claim must be considered in judging the patentability of that claim against the prior art." *In re Wilson*, 424 F.2d 1382, 1385, 165 USPQ 494, 496 (CCPA 1970).

1. Claims 22-23

With regard to independent claims 22-23, the references fail to teach or suggest all elements of these claims. These independent claims, like independent claim 25, also recite "retrieving one or more translation files derived from the management data, each of the translation files corresponding to at least one national language." As with independent claim 25, the Examiner relies on col. 5, lines 43-46 of Agnihotri to support the proposition that the prior art teaches or suggests this feature. Appellants have shown, in the preceding section of this Response, that Agnihotri fails to teach or suggest this feature, however. Moreover, Appellants respectfully submit that the CIM reference also fails to teach or suggest this feature.

In addition, claims 22 and 23 also recite the features of retrieving one or more plug-in code files derived from the management data and retrieving one or more display panel files derived from the management data. For the reasons outlined with respect to claims 1, 8, 15, and 25 in the previous section of this Response, Agnihotri fails to teach or suggest these features. Appellants respectfully submit that the CIM reference also fails to teach or suggest these features.

Therefore, the Office's burden of establishing a *prima facie* case of obviousness with respect to claims 22-23 as not been met. Accordingly, Appellants respectfully request that the Board ***REVERSE*** the Examiner's rejection of these claims.

2. Claim 24

Claim 24 recites a feature of "code generation logic for generating a console plug-in code file for each of the console identifiers." This feature is neither taught nor suggested by the cited references. As stated in the previous section of this Response, to the extent that one can analogize the "applet components" of Agnihotri to the "plug-in code files" recited in the presently rejected claims (as the Examiner has attempted to do), Agnihotri fails to teach or suggest anything regarding the origins of those "applet components." Thus, Agnihotri certainly fails to teach or suggest a feature of *generating* a console plug-in code file, as recited in claim 24.

In addition, claim 24 also recites the features of retrieving one or more plug-in code files derived from the management data and retrieving one or more display panel files derived from the management data. For the reasons outlined with respect to claims 22 and 23 above, neither Agnihotri nor CIM teaches or suggests these features.

Therefore, the Office's burden of establishing a *prima facie* case of obviousness with respect to claim 24 has not been met and Appellants respectfully request that the Board ***REVERSE*** these rejections.

3. Claims 2, 9, and 16

Claims 2, 9, and 16 are dependent claims that depend on independent claims 1, 8, and 15. Appellants have already demonstrated claims 1, 8, and 15 to be in condition for allowance. Appellants respectfully submit that claims 2, 9, and 16 are also allowable, at least by virtue of their dependency on allowable claims.

For the foregoing reasons, Appellants submit that claims 2, 9, 16, and 22-24 are not obvious and are therefore patentable over the references. Accordingly, Appellants respectfully request that the Examiner's rejections of claims 2, 9, 16, and 22-24 be ***REVERSED***.

Conclusion

For the foregoing reasons, Appellants submit that Claims 1, 3-8, 10-15, 17-21 and 25 over Agnihotri and respectfully request that the Board reverse these rejections. Likewise, claims 2, 9, 16, and 22-24 are allowable over Agnihotri in view the CIM reference and Appellants respectfully request that the Board reverse these rejections.

Respectfully submitted,

By _____

Joseph T. Van Leeuwen, Reg. No. 44,383
Van Leeuwen & Van Leeuwen
Attorneys for Appellants
Telephone: (512) 301-6738
Facsimile: (512) 301-6742

I. APPENDIX OF CLAIMS

1. A method of packaging management data adapted to interoperate with one or more management consoles, said method comprising:
   receiving one or more console identifiers, each of the console identifiers corresponding to one of the management consoles;
   retrieving one or more plug-in code files, each of the plug-in code files derived from the management data and each adapted to interface with one of the management consoles;
   retrieving one or more display panel files derived from the management data; and
   writing the plug-in code files and the display panels to a distribution medium.

2. The method as described in claim 1 wherein the management data includes a common information model managed object format file.

3. The method as described in claim 1 further comprising:
   retrieving one or more translation files derived from the management data, each of the translation files corresponding to at least one national language; and
   writing the translation files to the distribution medium.

4. The method as described in claim 1 wherein each of the display panel files is adapted to operate with a plurality of the management consoles.

5. The method as described in claim 1 further comprising:
   retrieving one or more plug-in runtime algorithms, each of the algorithms corresponding to one of the console identifiers;
   generating a console plug-in code file for each of the console identifiers; and
   compiling each of the generated console plug-in files, the compiling resulting in an executable entity adapted to interface with the management console corresponding to the console identifier.

6. The method as described in claim 1 further comprising:

loading the distribution medium into a computer system;

displaying a name corresponding to each of the management consoles in a selection display;

receiving one or more selections from a user, each of the selections corresponding to one of the management consoles;

copying the plug-in code files corresponding to the selected management consoles from the distribution medium to a nonvolatile storage device accessible by the computer system;

copying the display panel files from the distribution medium to a nonvolatile storage device accessible by the computer system; and

registering each of the plug-in code files with one or more installed management consoles, wherein the installed management consoles are installed on the computer system.

7. The method as described in claim 6 further comprising:

invoking one of the installed management consoles;

receiving a console selection from a user; and

displaying a display panel corresponding to one of the display panel files in response to the received selection.

8. An information handling system comprising:

one or more processors;

a memory accessible by the processors;

a nonvolatile storage area accessible by the processors; and

a packaging tool for packaging management data adapted to interoperate with one or more management consoles, the packaging tool including:

input logic for receiving one or more console identifiers, each of the console identifiers corresponding to one of the management consoles;

retrieval logic for retrieving one or more plug-in code files, each of the plug-in code files derived from the management data and each adapted to interface with one of the management consoles;

retrieval logic for retrieving one or more display panel files derived from the management data; and

output logic for writing the plug-in code files and the display panels to a distribution medium.

9. The information handling system as described in claim 8 wherein the management data includes a common information model managed object format file.

10. The information handling system as described in claim 8 further comprising:

retrieval logic for retrieving one or more translation files derived from the management data, each of the translation files corresponding to at least one national language; and

output logic for writing the translation files to the distribution medium.

11. The information handling system as described in claim 8 wherein each of the display panel files is adapted to operate with a plurality of the management consoles.

12. The information handling system as described in claim 8 further comprising:

retrieval logic for retrieving one or more plug-in runtime algorithms, each of the algorithms corresponding to one of the console identifiers;

code generation logic for generating a console plug-in code file for each of the console identifiers; and

compiler logic for compiling each of the generated console plug-in files, the compiling resulting in an executable entity adapted to interface with the management console corresponding to the console identifier.

13. The information handling system as described in claim 8 further comprising:

install logic for loading the distribution medium into a second information handling system;

display logic executed on the second information handling system for displaying a name corresponding to each of the management consoles in a selection display;

input logic executed on the second information handling system for receiving one or more selections from a user, each of the selections corresponding to one of the management consoles;

copy logic executed on the second information handling system for copying the plug-in code files corresponding to the selected management consoles from the distribution medium to a nonvolatile storage device accessible by the computer system;

copy logic executed on the second information handling system for copying the display panel files from the distribution medium to a nonvolatile storage device accessible by the computer system; and

console registration logic executed on the second information handling system for registering each of the plug-in code files with one or more installed management consoles, wherein the installed management consoles are installed on the computer system.

14. The information handling system as described in claim 13 further comprising:
logic for invoking one of the installed management consoles on the second information handling system;
input logic for receiving a console selection from a user; and
a display panel displayed on a display device, the display panel corresponding to one of the display panel files in response to the received selection.

15. A computer program product stored on a computer operable medium for packaging management data adapted to interoperate with one or more management consoles, said computer program product comprising:
means for receiving one or more console identifiers, each of the console identifiers corresponding to one of the management consoles;
means for retrieving one or more plug-in code files, each of the plug-in code files derived from the management data and each adapted to interface with one of the management consoles;
means for retrieving one or more display panel files derived from the management data; and
means for writing the plug-in code files and the display panels to a distribution medium.

16. The computer program product as described in claim 15 wherein the management data includes a common information model managed object format file.

17. The computer program product as described in claim 15 further comprising:
means for retrieving one or more translation files derived from the management data, each of the translation files corresponding to at least one national language; and
means for writing the translation files to the distribution medium.

18. The computer program product as described in claim 15 wherein each of the display panel files is adapted to operate with a plurality of the management consoles.

19. The computer program product as described in claim 15 further comprising:

means for retrieving one or more plug-in runtime algorithms, each of the algorithms corresponding to one of the console identifiers;

means for generating a console plug-in code file for each of the console identifiers; and

means for compiling each of the generated console plug-in files, the compiling resulting in an executable entity adapted to interface with the management console corresponding to the console identifier.

20. The computer program product as described in claim 15 further comprising:

means for loading the distribution medium onto a computer system;

means for displaying a name corresponding to each of the management consoles in a selection display;

means for receiving one or more selections from a user, each of the selections corresponding to one of the management consoles;

means for copying the plug-in code files corresponding to the selected management consoles from the distribution medium to a nonvolatile storage device accessible by the computer system;

means for copying the display panel files from the distribution medium to a nonvolatile storage device accessible by the computer system; and

means for registering each of the plug-in code files with one or more installed management consoles, wherein the installed management consoles are installed on the computer system.

21. The computer program product as described in claim 20 further comprising:

means for invoking one of the installed management consoles;

means for receiving a console selection from a user; and

means for displaying a display panel corresponding to one of the display panel files in response to the received selection.

22. A method of packaging management data adapted to interoperate with one or more management consoles, wherein the management data includes a common information model managed object format file, said method comprising:

receiving one or more console identifiers, each of the console identifiers corresponding to one of the management consoles;

retrieving one or more plug-in code files, each of the plug-in code files derived from the management data and each adapted to interface with one of the management consoles;

retrieving one or more display panel files derived from the management data;

retrieving one or more translation files derived from the management data, each of the translation files corresponding to at least one national language; and

writing the translation files, the plug-in code files and the display panels to a distribution medium.

23. A method of packaging management data adapted to interoperate with one or more management consoles, wherein the management data includes a common information model managed object format file, said method comprising:

receiving one or more console identifiers, each of the console identifiers corresponding to one of the management consoles;

retrieving one or more plug-in code files, each of the plug-in code files derived from the management data and each adapted to interface with one of the management consoles;

retrieving one or more display panel files derived from the management data;

retrieving one or more translation files derived from the management data, each of the translation files corresponding to at least one national language;

retrieving one or more plug-in runtime algorithms, each of the algorithms corresponding to one of the console identifiers;

generating a console plug-in code file for each of the console identifiers;

compiling each of the generated console plug-in files, the compiling resulting in an executable entity adapted to interface with the management console corresponding to the console identifier;

writing the translation files, the compiled plug-in code files and the display panels to a distribution medium.

24. An information handling system comprising:

one or more processors;

a memory accessible by the processors;

a nonvolatile storage area accessible by the processors; and

a packaging tool for packaging management data adapted to interoperate with one or more management consoles, wherein the management data includes a common information model managed object format file, the packaging tool including:

input logic for receiving one or more console identifiers, each of the console identifiers corresponding to one of the management consoles;

retrieval logic for retrieving one or more plug-in code files, each of the plug-in code files derived from the management data and each adapted to interface with one of the management consoles;

retrieval logic for retrieving one or more plug-in runtime algorithms, each of the algorithms corresponding to one of the console identifiers;

code generation logic for generating a console plug-in code file for each of the console identifiers; and

compiler logic for compiling each of the generated console plug-in files, the compiling resulting in an executable entity adapted to interface with the management console corresponding to the console identifier;

retrieval logic for retrieving one or more display panel files derived from the management data;

output logic for writing the compiled plug-in code files and the display panels to a distribution medium.

25. A computer program product stored on a computer operable medium for packaging management data adapted to interoperate with one or more management consoles, said computer program product comprising:

means for receiving one or more console identifiers, each of the console identifiers corresponding to one of the management consoles;

means for retrieving one or more plug-in code files, each of the plug-in code files derived from the management data and each adapted to interface with one of the management consoles;

means for retrieving one or more display panel files derived from the management data;

means for retrieving one or more translation files derived from the management data, each of the translation files corresponding to at least one national language; and

means for writing the translation files the plug-in code files and the display panels to a distribution medium.

J. EVIDENCE APPENDIX

Not applicable.

K. RELATED PROCEEDINGS APPENDIX

None. Note that two related applications are currently being appealed (10/047,792 and 10/046,940), however no decisions have yet been rendered by a court or the Board on either of these appeals.